Internal Architecture of 8086

Internal Architecture of 8086

Divided into two independent functional units.

- Bus Interface Unit
- Execution Unit

Bus Interface Unit (BIU):

It provides a full 16-bit bidirectional data bus and 20-bit address bus. The bus interface unit is responsible for performing all external bus operations. Specifically it has the following functions – instruction fetch, instruction queuing, operand fetch and storage, address relocation and bus control. The BIU uses a mechanism known as an instruction stream queue to implement a pipeline architecture.

Bus Interface Unit (BIU):

Functions –

i. Fetch the instruction or data from memory.

ii. Write the data to memory.

iii. Write the data to ports.

iv. Read data from ports.

Bus Interface Unit (BIU) divided into three functional parts.

- Instruction Pointer (IP)
- Instruction Queue
- Segment Register

Internal Architecture of 8086 (Cont.)

Instruction Pointer (IP): It is a 16-bit register that keeps the address of memory location of coming instruction to be executed. So, IP holds the address of the next instruction to be executed.

Instruction Queue: BIU performs its operation in parallel with Execution Unit (EU). BIU fetches instruction byte while execution unit is executing operations. The prefetched instruction is saved in group of high speed registers is known as instruction queue.

Internal Architecture of 8086 (Cont.)

Segment Register: The memory space 1MB of 8086 is segment into 4 blocks. Each block specified by register with maximum size 64KB. 4 blocks are – Code segment (CS), Data segment (DS), Stack segment (SS) and Extra segment (ES).

Code segment (CS): Used for addressing a memory location where the programs are stored.

Data segment (DS): It contains data which is used by the program.

Stack segment (SS): Defined as area of memory used for stack.

Extra segment (ES): Similar to data segment where additional data is stored.

Internal Architecture of 8086 (Cont.)

- This queue permits prefetch up to six bytes of instruction code. Whenever the queue of the BIU is not full, it has room for at least two more bytes and at the same time the EU is not requesting it to read or write operands from memory, the BIU is free to look ahead in the program by prefetching the next sequential instruction.
- These prefetching instructions are held in its FIFO queue. With its 16-bit data bus, the BIU fetches two instruction bytes in a single memory cycle.
- After a byte is loaded at the input end of the queue, it automatically shifts up through the FIFO to the empty location nearest the output.

Internal Architecture of 8086 (Cont.)

- If the BIU is already in the process of fetching an instruction when the EU request it to read or write operands from memory or I/O, the BIU first completes the instruction fetch bus cycle before initiating the operand read/write cycle.
- The BIU also contains a dedicated adder which is used to generate the 20 bit physical address that is output on the address bus. This address is formed by adding an appended 16 bit segment address and a 16 bit offset address.
- For example, the physical address of the next instruction to be fetched is formed by combining the current contents of the code segment CS register and the current contents of the instruction pointer IP register.
- The BIU is also responsible for generating bus control signals such as those for memory read or write and I/O read or write.

Execution Unit (EU):

- The execution unit is responsible for decoding and executing all instructions.
- The EU extracts instructions from the top of the queue in the BIU, decodes them, generates operands if necessary, passes them to the BIU and requests it to perform the read or write bus cycles to memory or I/O and perform the operation specified by the instruction on the operands.
- During the execution of the instruction, the EU tests the status and control flags and updates them based on the results of executing the instruction.

Functions –

i. To tell BIU where to fetch the instruction or data from.

ii. To decode the instruction.

iii. To execute the instruction.

iv. EU contains the control circuitry to perform various internal operations.

Execution Unit (EU):

Execution Unit (EU) divided into five functional parts -

- General Purpose Registers (GPRs)
- Pointer and Indexed Registers
- ALU
- Flag Register
- Timing and Control Unit

General Purpose Register:

8086 Microprocessor consists of 4 GPR.

AX: Accumulator register, stores the result.

BX: Base Register, store the starting base location of the memory.

CX: Counter register, used in loop instruction.

DX: Data register, used to contain I/O port address for I/O instruction.

These all are16-bit register. AH, AL, BH, BL, CH, CL, DH, DL are 8-bit register.

Pointer and Indexed Register: 8086 has two pointers and two indexed registers. Stack Pointer (SP), Base Pointer (BP), Source Index (SI) and Destination Index (DI).

ALU: 16-bit and it performs arithmetic and logical operations of 8-bit and as well as 16-bit.

Flag Register: 8086 Microprocessor has 16-bit flag register. (9 flags are there and 7 bits unused. Among 9 flags we divided into 2 categories. 6 Status flags and 3 control flags. 6 status flag are carry flag, auxiliary carry flag, zero flag, sign flag, parity flag and overflow flag. 3 control flags are direction flag, Interrupt Enable flag and Trap flag.

D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0
X	X	X	X	OF	DF	IF	TF	SF	ZF	X	AC	X	PF	X	CF

Trap flag: TF=0 if it is set (TF=0) the processor will enter in single step execution mode.

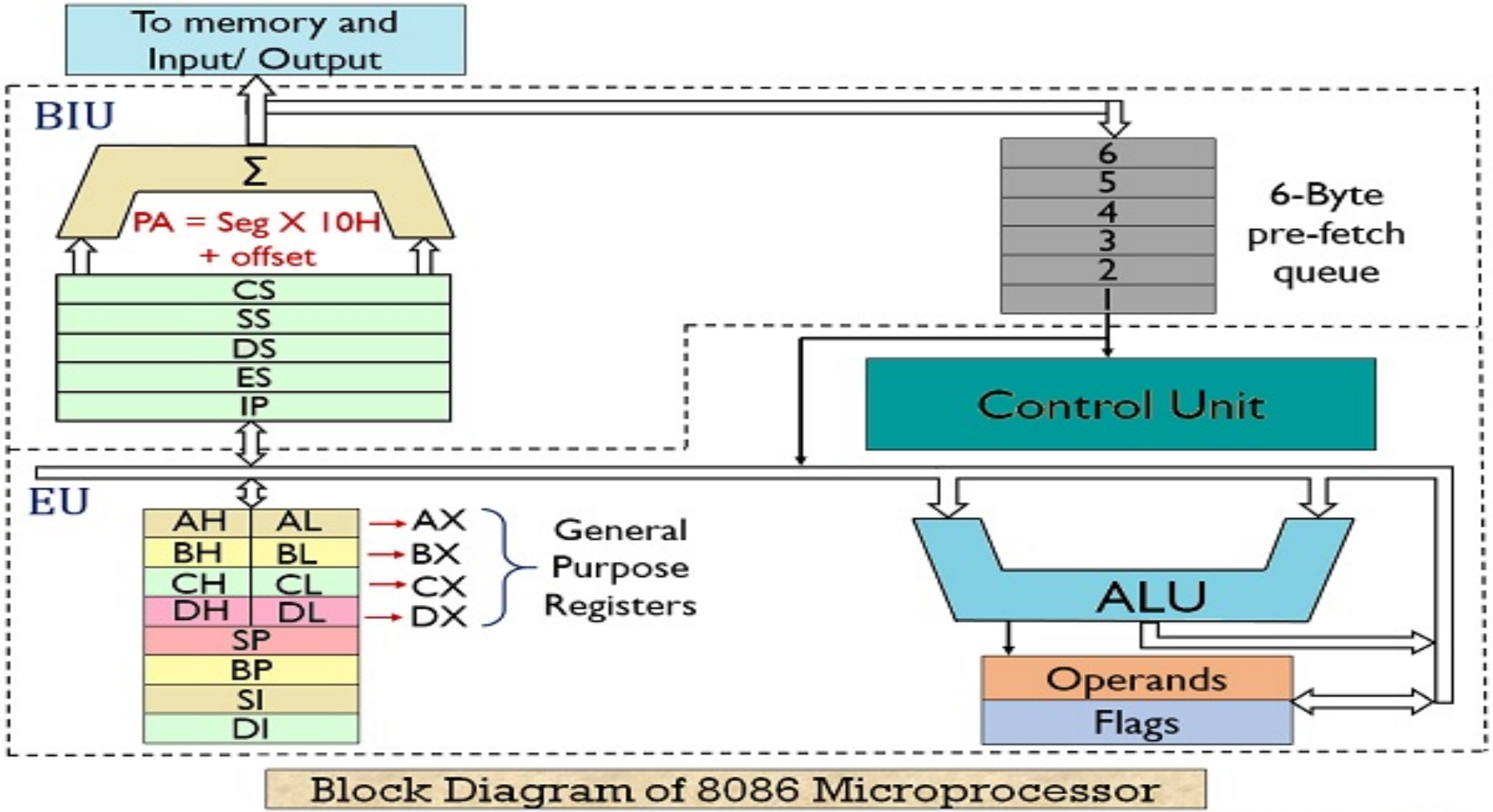
Interrupt flag: IF=0, maskable interrupts are recognized.

Direction flag: Used for string manipulation instruction.

If DF=0, Auto increment mode, DF=1, Auto decrement mode.

Overflow flag: OF=0, if the result is too large positive number or too small negative number to fit in accumulator.

Timing and Control Unit: The control unit of execution unit directs all internal operations and also responsible for generation of control signals.



Block Diagram of 8086 Microprocessor